

# Examen

## 105000016 - Programación para Sistemas Grado en Ingeniería Informática (2009)

Lenguajes y Sistemas Informáticos e Ingeniería de Software

Facultad de Informática

Universidad Politécnica de Madrid

Curso 2010/2011 - Junio 2011

### Normas

- El examen puntúa sobre **12 puntos**.
- La duración total del mismo es de **una hora y cuarto**.
- Se deberá tener el DNI o el carnet de la UPM en lugar visible.
- No olvidar rellenar **apellidos, nombre y número de matrícula** en cada hoja.
- La solución al examen se proporcionará antes de la revisión.
- Las calificaciones se darán a conocer el **29 de junio** a través del Moodle de la asignatura.
- La revisión del examen tendrá lugar el **30 de junio** a las 10:00 en la sala 2319.

### Cuestionario

- (1 punto) 1. Escribir el *makefile* que permita compilar una aplicación que consta de 2 archivos fuentes `leer.c` y `dibujar.c` y un archivo cabecera `dibujar.h` donde están las declaraciones de todas las funciones usadas por `dibujar.c`. La función `main` está incluida en `dibujar.c`. La aplicación usa una biblioteca del sistema denominada `libjpeg.a`. El nombre del ejecutable será `dibuja`.

#### Solución:

```
dibuja:  dibujar.o readjpeg.o
        gcc -Wall -o dibujar dibujar.o readjpeg.o -ljpeg
dibujar.o: dibujar.c dibujar.h
        gcc -Wall -c dibujar.c
readjpeg.o: readjpeg.c
        gcc -Wall -c readjpeg.c
```

(1 punto) 2. Escriba la salida que genera el siguiente programa en C:

```
#include <stdio.h>
int main (int argc, char*argv[])
{

    int lista[]= {1,2,3,4,5};
    int *p;
    p=&lista[0];
    printf("*p=_ %d\n", *(p+1));
    printf("*P=_ %d\n", p[2]);

    return 0;
}
```

**Solución:**

```
*p = 2
*P = 3
```

(1 punto) 3. Escriba la salida que genera el siguiente programa en C :

```
#include <stdio.h>

int main (int argc, char*argv[])
{
    int x=11;

    while (x>0) {
        printf("%d_", x);
        x=x/2;
    }

    return 0;
}
```

**Solución:**

```
11 5 2 1
```

(1 punto) 4. Escribir una única instrucción que permitiría en C hacer una reserva de memoria dinámica para 300 enteros y todas estas posiciones se inicialicen con el valor 0.

**Solución:**

```
p = (int *) calloc(300, sizeof(int));
```

(1 punto) 5. Escriba la salida que genera el siguiente programa en C:

```
#include <stdio.h>

int main (int argc, char*argv[])
{

    int i=0, v=4;
    while (i<v) {

        switch(i)
        {

            case 1: printf("A\n");
            case 2: printf("B\n");break;
            case 3: printf("C"); break;
            default: printf("D");
        }
        i++;
    }

    return 0;

}
```

**Solución:**

DA  
B  
B  
C

(1 punto) 6. Escribir una definición de tipos (**typedef**) y la correspondiente declaración de las variables a para que las siguientes operaciones tengan sentido:

```
a[5].longitud=34.76;
a[2].frase[9] = '\0';
```

**Solución:**

```
typedef struct
{
    double longitud;
    char frase [10];
} T_A;
T_A a[6];
```

- (1 punto) 7. Se está realizando un programa prog que tiene prog.c como fichero fuente asociado. El ejecutable ha dado un error de ejecución y se quiere llamar al depurador GDB con un *core* para intentar localizar dónde se produce el error. Indique todas las acciones, especificando las llamadas concretas a compilador, sistema operativo, depurador, etc.. que debe realizar para ello.

**Solución:**

```
gcc -g -Wall -ansi -pedantic prog.c -o prog
ulimit -c unlimited
./prog
gdb prog core
```

- (1 punto) 8. Escriba la salida que genera el siguiente programa en C:

```
#include <stdio.h>

int main (int argc, char*argv[])
{
    int x, y;
    int *p1, *p2;
    p1= &x;
    y=7;
    y=(*p1);
    x=3;
    printf(" %d_", *p1);
    p2=&y;
    (*p2)=(*p1)*2;
    (*p1)=23;
    printf(" %d_%d", y, x);
}
```

**Solución:**

```
3 6 23
```

(1 punto) 9. ¿Cuál es el caracter que indica el inicio de un comentario?

- A. %
- B. #**
- C. !
- D. -

(1 punto) 10. Señalar cuál de las siguientes expresiones Bash ejecuta la secuencia de comandos *list* dentro de un subentorno no afectando al entorno actual.

- A. { *list* }
- B. ( *list* )**
- C. (( *list* ))

(1 punto) 11. ¿Existe alguna diferencia entre \$(ls) y `ls`?

- A. Si.
- B. No.**

(1 punto) 12. ¿Cuál de los siguientes comandos comprueba si el fichero /etc/passwd es legible?

- A. **test** -a /etc/passwd
- B. [ -r /etc/passwd ]**
- C. **test** -L /etc/passwd
- D. [ /etc/passwd ]